

CLAIMS

- 1 1. A method of protecting computer data, said method comprising the steps of:
2 acquiring a first state snapshot S_1 of a first data volume consistent state at a
3 time t_1 ;
4 acquiring a second state snapshot S_2 of a second data volume consistent state
5 at a time $t_2 > t_1$;
6 generating a first precedent snapshot difference list S_{21} comprising an
7 identification of data blocks of said first state snapshot S_1 differing
8 from data blocks in said second state snapshot S_2 ; and
9 creating a first precedent backup B_{21} by copying from said first state
10 snapshot S_1 data blocks identified in said first precedent snapshot
11 difference list S_{21} , said first precedent backup B_{21} further comprising
12 said first precedent snapshot difference list S_{21} .
- 1 2. The method of claim 1 further comprising the step of storing said first precedent
2 backup B_{21} in an offline memory means.
- 1 3. The method of claim 2 wherein said offline memory means comprises at least one
2 member from the group consisting of magnetic tape and optical disk.
- 1 4. The method of claim 1 further comprising the step of deleting said first state
2 snapshot S_1 following said step of generating a first precedent snapshot difference list
3 S_{21} .

1 5. The method of claim 2 further comprising the steps of:

2 retrieving said first precedent backup B_{21} ;
3 recovering said first precedent snapshot difference list S_{21} from said first
4 precedent backup B_{21} ; and
5 restoring said first state snapshot S_1 by overwriting at least a portion of said
6 second state snapshot S_2 with the contents of said first precedent
7 backup B_{21} .

1 6. The method of claim 1 further comprising the steps of:

2 acquiring a third state snapshot S_3 of a third data volume consistent state at a
3 time $t_3 > t_2$;
4 generating a second precedent snapshot difference list S_{32} comprising an
5 identification of data blocks of said second state snapshot S_2 differing
6 from data blocks in said third state snapshot S_3 ; and
7 creating a second precedent backup B_{32} by copying from said second state
8 snapshot S_2 data blocks identified in said second precedent snapshot
9 difference list S_{32} , said second precedent backup B_{32} further
10 comprising said second precedent snapshot difference list S_{32} .

1 7. The method of claim 6 further comprising the step of storing said second
2 precedent backup B_{32} in an offline memory means.

1 8. The method of claim 6 further comprising the step of deleting said second state
2 snapshot S_2 following said step of generating a second precedent snapshot difference list
3 S_{32} .

1 9. The method of claim 6 further comprising the steps of:
2 recovering said second precedent snapshot difference list S_{32} from said
3 second precedent backup B_{32} ; and
4 restoring said second state snapshot S_2 by overwriting at least a portion of
5 said third state snapshot S_3 with the contents of said second precedent
6 backup B_{32} .

1 10. The method of claim 6 further comprising the steps of:
2 generating a concatenated precedent snapshot difference list S_{31} comprising
3 an identification of said data blocks of said second state snapshot S_2
4 differing from data blocks in said third state snapshot S_3 and an
5 identification of said data blocks of said first state snapshot S_1 differing
6 from data blocks in said second state snapshot S_2 ;
7 creating a concatenated backup B_{31} by copying all said blocks in said first
8 precedent backup B_{21} and copying all blocks in said second precedent
9 backup B_{32} not present in said first precedent backup B_{21} ;
10 copying said first precedent snapshot difference list S_{21} and said second
11 precedent snapshot difference list S_{32} into said concatenated backup
12 B_{31} ; and
13 storing said concatenated backup B_{31} in an offline memory means.

- 1 11. The method of claim 10 further comprising the steps of:
2 retrieving said concatenated backup B_{31} ;
3 recovering said concatenated precedent snapshot difference list S_{31} from said
4 concatenated backup B_{31} ; and
5 restoring said first state snapshot S_1 by overwriting at least a portion of said
6 third state snapshot S_3 with the contents of said concatenated backup
7 B_{31} .
- 1 12. The method of claim 10 further comprising the steps of:
2 acquiring a fourth state snapshot S_4 of a fourth data volume consistent state
3 at a time $t_4 > t_3$;
4 generating a third precedent snapshot difference list S_{43} comprising an
5 identification of data blocks of said third state snapshot S_3 differing
6 from data blocks in said fourth state snapshot S_4 ;
7 creating a third precedent backup B_{43} by copying from said third state
8 snapshot S_3 data blocks identified in said third precedent snapshot
9 difference list S_{43} , said third precedent backup B_{43} further comprising
10 said third precedent snapshot difference list S_{43} .
11 generating a concatenated precedent snapshot difference list S_{42} comprising
12 an identification of said data blocks of said third state snapshot S_3
13 differing from data blocks in said fourth state snapshot S_4 and an
14 identification of said data blocks of said second state snapshot S_2
15 differing from data blocks in said third state snapshot S_3 ;
16 creating a concatenated backup B_{42} by copying all said blocks in said second
17 precedent backup B_{32} and copying all blocks in said third precedent
18 backup B_{43} not present in said second precedent backup B_{32} , and

19 copying said second and third precedent difference lists S_{32} and S_{43}
20 into said concatenated backup B_{42} ; and
21 creating a concatenated backup B_{41} by copying all said blocks in said first
22 precedent backup B_{21} and copying all blocks in said concatenated
23 backup B_{42} not present in said first precedent backup B_{21} , and copying
24 first, second, and third precedent difference lists S_{21} , S_{32} , and S_{43} , into
25 said concatenated backup B_{41} .

1 13. The method of claim 12 further comprising the steps of:
2 retrieving said concatenated backup B_{41} ;
3 recovering said concatenated precedent snapshot difference list S_{41} from said
4 concatenated backup B_{41} ; and
5 restoring said first state snapshot S_1 by overwriting at least a portion of said
6 fourth state snapshot S_4 with the contents of said concatenated backup
7 B_{41} .

1 14. A method of protecting computer data, said method comprising the steps of:
2 acquiring a base state snapshot S_0 of a data volume base state at a time t_0 ;
3 creating a base state backup B_0 of said base state snapshot S_0 ;
4 acquiring a first state snapshot S_1 of a first data volume consistent state at a
5 time $t_1 > t_0$;
6 generating a first succedent snapshot difference list S_{01} comprising an
7 identification of data blocks of said first state snapshot S_1 differing
8 from data blocks in said base state snapshot S_0 ;
9 creating a first succedent backup B_{01} by copying from said first state
10 snapshot S_1 data blocks identified in said first succedent snapshot

11 difference list S_{01} , and copying said first succedent snapshot difference
12 list S_{01} ; and
13 deleting at least a portion of said first state snapshot S_1 .

1 15. The method of claim 14 further comprising the step of storing said base state
2 backup B_0 and said first succedent backup B_{01} in an offline memory means.

1 16. The method of claim 15 further comprising the steps of:
2 retrieving said base state backup B_0 and said first succedent backup B_{01} ;
3 recovering said base state snapshot S_0 and said first succedent snapshot
4 difference list S_{01} from said base state backup B_0 and said first
5 succedent backup B_{01} , respectively; and
6 restoring said first state snapshot S_1 by overwriting said base state snapshot
7 S_0 with said first succedent backup B_{01} .

1 17. The method of claim 14 further comprising the steps of:
2 acquiring a second state snapshot S_2 of a second data volume consistent state
3 at a time $t_2 > t_1$;
4 generating a second succedent snapshot difference list S_{12} comprising an
5 identification of data blocks of said second state snapshot S_2 differing
6 from data blocks in said first state snapshot S_1 ;
7 creating a second succedent backup B_{12} by copying from said second state
8 snapshot S_2 data blocks identified in said second succedent snapshot
9 difference list S_{12} , and copying said second succedent snapshot
10 difference list S_{12} ; and
11 deleting at least a portion of said second state snapshot S_2 .

1 18. A method of protecting computer data, said method comprising the steps of:

2 acquiring a base state snapshot S_0 of a data volume base state at a time t_0 ;

3 acquiring a first state snapshot S_1 of a first data volume consistent state at a

4 time $t_1 > t_0$;

5 acquiring a second state snapshot S_2 of a second data volume consistent state

6 at a time $t_2 > t_1$;

7 generating a first succedent snapshot difference list S_{01} comprising an

8 identification of data blocks of said first state snapshot S_1 differing

9 from data blocks in said base state snapshot S_0 ;

10 generating a first precedent snapshot difference list S_{21} comprising an

11 identification of data blocks of said first state snapshot S_1 differing

12 from data blocks in said second state snapshot S_2 ;

13 creating a first composite backup C_{012} by copying from said first state

14 snapshot S_1 data blocks identified in first succedent snapshot

15 difference list S_{01} and copying from said first state snapshot S_1 data

16 blocks identified in said first precedent snapshot difference list S_{21} ;

17 and

18 copying said first succedent snapshot difference list S_{01} and said first

19 precedent snapshot difference list S_{21} into said first composite backup

20 C_{012} .

1 19. The method of claim 18 further comprising the step of storing said first composite

2 backup C_{012} in an offline memory means.

1 20. The method of claim 19 further comprising the steps of:

2 retrieving said first composite backup C_{012} ;
3 recovering said first precedent snapshot difference list S_{21} ; and
4 restoring said first state snapshot S_1 by overwriting at least a portion of said
5 second state snapshot S_2 with at least a portion of the contents of said
6 first composite backup C_{012} .

1 21. The method of claim 18 further comprising the steps of:

2 acquiring a third state snapshot S_3 of a third data volume consistent state at a
3 time $t_3 > t_2$;
4 generating a second succedent snapshot difference list S_{12} comprising an
5 identification of data blocks of said second state snapshot S_2 differing
6 from data blocks in said first state snapshot S_1 ;
7 generating a second precedent snapshot difference list S_{32} comprising an
8 identification of data blocks of said second state snapshot S_2 differing
9 from data blocks in said third state snapshot S_3 ;
10 creating a second composite backup C_{123} by copying from said second state
11 snapshot S_2 data blocks identified in second succedent snapshot
12 difference list S_{12} and copying from said second state snapshot S_2 data
13 blocks identified in said second precedent snapshot difference list S_{32} ;
14 and
15 copying said second succedent snapshot difference list S_{12} and said second
16 precedent snapshot difference list S_{32} into said second composite
17 backup C_{123} .

1 22. The method of claim 21 further comprising the step of storing said second
2 composite backup C_{123} in an offline memory means.

1 23. The method of claim 18 further comprising the steps of:

2 acquiring a plurality of third through n^{th} state snapshots S_3 through S_n of

3 third through n^{th} data volume consistent states at respective times

4 $t_3 < \dots t_j \dots \leq t_n$;

5 generating second through $(n-1)^{\text{th}}$ succedent snapshot difference lists S_{12}

6 through $S_{(n-2)(n-1)}$ respectively, a $(j-1)^{\text{th}}$ succedent snapshot

7 difference list $S_{(j-2)(j-1)}$ comprising an identification of data blocks of a

8 $(j-1)^{\text{th}}$ state snapshot $S_{(j-1)}$ differing from data blocks in a $(j-2)^{\text{th}}$

9 state snapshot $S_{(j-2)}$;

10 generating second through $(n-1)^{\text{th}}$ precedent snapshot difference lists S_{32}

11 through $S_{(n)(n-1)}$ respectively, a $(j-1)^{\text{th}}$ precedent snapshot difference

12 list $S_{(j)(j-1)}$ comprising an identification of data blocks of a $(j-1)^{\text{th}}$

13 state snapshot $S_{(j-1)}$ differing from data blocks in a j^{th} state snapshot

14 S_j ;

15 creating second through $(n-1)^{\text{th}}$ composite backups C_{123} through

16 $C_{(n-2)(n-1)(n)}$, wherein a $(j-1)^{\text{th}}$ composite backup $C_{(j-2)(j-1)(j)}$ is

17 created by copying from said $(j-1)^{\text{th}}$ state snapshot $S_{(j-1)}$ data blocks

18 identified in said $(j-1)^{\text{th}}$ succedent snapshot difference list $S_{(j-2)(j-1)}$

19 and copying from said $(j-1)^{\text{th}}$ state snapshot $S_{(j-1)}$ data blocks

20 identified in said $(j-1)^{\text{th}}$ precedent snapshot difference list $S_{(j)(j-1)}$;

21 and

22 copying said succedent snapshot difference lists S_{12} through $S_{(n-2)(n-1)}$ and

23 said precedent snapshot difference lists S_{32} through $S_{(n)(n-1)}$ into said

24 respective composite backups C_{123} through $C_{(n-2)(n-1)(n)}$.

1 24. The method of claim 23 further comprising the step of storing one or more of said
2 second through $(n-1)^{\text{th}}$ composite backups C_{123} through $C_{(n-2)(n-1)(n)}$ in an offline
3 memory means.

1 25. The method of claim 23 further comprising the step of deleting said third through
2 $(n-1)^{\text{th}}$ state snapshots S_3 through S_{n-1} .

1 26. The method of claim 23 further comprising the steps of:
2 assigning a unique identifier to each said state snapshot S_j ;
3 for each said composite backup $C_{(j-2)(j-1)(j)}$, identifying said state snapshots
4 $S_{(j-2)}$ and S_j as difference snapshots and said $(j-1)^{\text{th}}$ state snapshot
5 $S_{(j-1)}$ as a content snapshot;
6 for each said composite backup $C_{(j-2)(j-1)(j)}$, copying the unique identifiers of
7 said state snapshots $S_{(j-2)}$, $S_{(j-1)}$, and S_j into said $(j-1)^{\text{th}}$ composite
8 backup $C_{(j-2)(j-1)(j)}$.

1 27. The method of claim 26 further comprising the steps of:
2 retrieving said $(j-1)^{\text{th}}$ composite backup $C_{(j-2)(j-1)(j)}$;
3 recovering said $(j-1)^{\text{th}}$ precedent snapshot difference list $S_{(j)(j-1)}$ from said
4 $(j-1)^{\text{th}}$ composite backup $C_{(j-2)(j-1)(j)}$; and
5 restoring said $(j-1)^{\text{th}}$ state snapshot S_{j-1} by overwriting at least a portion of
6 said j^{th} state snapshot S_j with the contents of said $(j-1)^{\text{th}}$ composite
7 backup $C_{(j-2)(j-1)(j)}$.

1 28. The method of claim 26 further comprising the step of comparing the unique
2 identifier of said j^{th} snapshot S_j to the unique identifiers of the difference snapshots of
3 said $(j-1)^{\text{th}}$ composite backup $C_{(j-2)(j-1)(j)}$.

1 29. The method of claim 26 further comprising the step of assigning the unique
2 identifier of the content snapshot of said $(j-1)^{\text{th}}$ composite backup $C_{(j-2)(j-1)(j)}$ to be the
3 unique identifier of said restored $(j-1)^{\text{th}}$ state snapshot $S_{(j-1)}$.

1 30. The method of claim 23 further comprising the steps of:

2 creating a $(j-1)^{\text{th}}$ precedent backup $B_{(j)(j-1)}$ by copying from said $(j-1)^{\text{th}}$
3 state snapshot $S_{(j-1)}$ data blocks identified in said $(j-1)^{\text{th}}$ precedent
4 snapshot difference list $S_{(j)(j-1)}$, said $(j-1)^{\text{th}}$ precedent backup $B_{(j)(j-1)}$
5 further comprising said $(j-1)^{\text{th}}$ precedent snapshot difference list
6 $S_{(j)(j-1)}$;

7 creating a $(j-2)^{\text{th}}$ precedent backup $B_{(j-1)(j-2)}$ by copying from said
8 $(j-2)^{\text{th}}$ state snapshot $S_{(j-2)}$ data blocks identified in said $(j-2)^{\text{th}}$
9 precedent snapshot difference list $S_{(j-1)(j-2)}$, said $(j-2)^{\text{th}}$ precedent
10 backup $B_{(j-1)(j-2)}$ further comprising said $(j-2)^{\text{th}}$ precedent snapshot
11 difference list $S_{(j-1)(j-2)}$; and

12 creating a first concatenated precedent backup $B_{(j)(j-2)}$ from said $(j-1)^{\text{th}}$
13 precedent backup $B_{(j)(j-1)}$ and said $(j-2)^{\text{th}}$ precedent backup
14 $B_{(j-1)(j-2)}$ by copying all blocks in said $(j-2)^{\text{th}}$ precedent backup
15 $B_{(j-1)(j-2)}$ and by also copying all blocks in said $(j-1)^{\text{th}}$ precedent
16 backup $B_{(j)(j-1)}$ not present in said $(j-2)^{\text{th}}$ precedent backup $B_{(j-1)(j-2)}$
17 and by copying said precedent difference lists $S_{(j-1)(j-2)}$ and $S_{(j)(j-1)}$

18 from said precedent backups $B_{(j-1)(j-2)}$ and $B_{(j)(j-1)}$ into said
19 concatenated precedent backup $B_{(j)(j-2)}$.

1 31. The method of claim 30 further comprising the steps of:

2 creating a $(j-3)^{\text{th}}$ precedent backup $B_{(j-2)(j-3)}$ by copying from said
3 $(j-3)^{\text{th}}$ state snapshot $S_{(j-3)}$ data blocks identified in said $(j-3)^{\text{th}}$
4 precedent snapshot difference list $S_{(j-2)(j-3)}$, said $(j-3)^{\text{th}}$ precedent
5 backup $B_{(j-2)(j-3)}$ further comprising said $(j-3)^{\text{th}}$ precedent snapshot
6 difference list $S_{(j-2)(j-3)}$; and
7 creating a second concatenated precedent backup $B_{(j)(j-3)}$ from said first
8 concatenated precedent backup $B_{(j)(j-2)}$ and said $(j-3)^{\text{th}}$ precedent
9 backup $B_{(j-2)(j-3)}$ by copying all blocks in said $(j-3)^{\text{th}}$ precedent
10 backup $B_{(j-2)(j-3)}$ and by also copying all blocks in said first
11 concatenated precedent backup $B_{(j)(j-2)}$ not present in said $(j-3)^{\text{th}}$
12 precedent backup $B_{(j-2)(j-3)}$, and by copying said precedent difference
13 lists $S_{(j-2)(j-3)}$, $S_{(j-1)(j-2)}$, and $S_{(j)(j-1)}$ from said precedent backup
14 $B_{(j-2)(j-3)}$ and said concatenated precedent backup $B_{(j)(j-2)}$ into said
15 concatenated precedent backup $B_{(j)(j-3)}$.

1 32. The method of claim 31 further comprising the steps of:

2 retrieving a concatenated precedent backup $B_{(h)(g)}$, where $g < h \leq n$; and
3 restoring a g^{th} state snapshot S_g by overwriting an h^{th} state snapshot S_h
4 with said concatenated precedent backup $B_{(h)(g)}$.

1 33. The method of claim 23 further comprising the steps of:

2 creating a $(j-1)^{\text{th}}$ succedent backup $B_{(j-2)(j-1)}$ by copying from said $(j-1)^{\text{th}}$

3 state snapshot $S_{(j-1)}$ data blocks identified in said $(j-1)^{\text{th}}$ succedent

4 snapshot difference list $S_{(j-2)(j-1)}$, said $(j-1)^{\text{th}}$ succedent backup

5 $B_{(j-2)(j-1)}$ further comprising said $(j-1)^{\text{th}}$ succedent snapshot

6 difference list $S_{(j-2)(j-1)}$;

7 creating a j^{th} succedent backup $B_{(j-1)(j)}$ by copying from said j^{th} state

8 snapshot S_j data blocks identified in said j^{th} succedent snapshot

9 difference list $S_{(j-1)(j)}$, said j^{th} succedent backup j^{th} further

10 comprising said j^{th} succedent snapshot difference list $S_{(j-1)(j)}$; and

11 creating a first concatenated succedent backup $B_{(j-2)(j)}$ from said j^{th}

12 succedent backup $B_{(j-1)(j)}$ and said $(j-1)^{\text{th}}$ succedent backup

13 $B_{(j-2)(j-1)}$ by copying all blocks in said j^{th} succedent backup $B_{(j-1)(j)}$

14 and by also copying all blocks in said $(j-1)^{\text{th}}$ succedent backup

15 $B_{(j-2)(j-1)}$ not present in said j^{th} succedent backup $B_{(j-1)(j)}$, and by

16 copying said difference lists $S_{(j-1)(j)}$ and $S_{(j-2)(j-1)}$.

1 34. The method of claim 33 further comprising the steps of:

2 creating a $(j-2)^{\text{th}}$ succedent backup $B_{(j-3)(j-2)}$ by copying from said

3 $(j-2)^{\text{th}}$ state snapshot $S_{(j-2)}$ data blocks identified in said $(j-2)^{\text{th}}$

4 succedent snapshot difference list $S_{(j-3)(j-2)}$, said $(j-2)^{\text{th}}$ succedent

5 backup $B_{(j-3)(j-2)}$ further comprising said $(j-2)^{\text{th}}$ succedent snapshot

6 difference list $S_{(j-3)(j-2)}$; and

7 creating a second concatenated succedent backup $B_{(j-3)(j)}$ from said first
8 concatenated succedent backup $B_{(j-2)(j)}$ and said $(j-2)^{\text{th}}$ succedent
9 backup $B_{(j-3)(j-2)}$ by copying all blocks in said first concatenated
10 succedent backup $B_{(j-2)(j)}$ and by also copying all blocks in said
11 $(j-2)^{\text{th}}$ succedent backup $B_{(j-3)(j-2)}$ not present in said first
12 concatenated succedent backup $B_{(j-2)(j)}$, and copying said difference
13 lists $S_{(j-1)(j)}$, $S_{(j-2)(j-1)}$, and $S_{(j-3)(j-2)}$ from said first concatenated
14 succedent backup $B_{(j-2)(j)}$ and said $(j-2)^{\text{th}}$ succedent backup
15 $B_{(j-3)(j-2)}$.

1 35. An apparatus suitable for protecting the data volume in a computer system, said
2 apparatus comprising:

3 means for acquiring a sequence of state snapshots $S_0, \dots, S_j, \dots, S_n$ of the
4 data volume, each said state snapshot acquired at a respective time
5 $t_0 < \dots t_j \dots < t_n$;
6 means for generating a $(j-1)^{\text{th}}$ precedent snapshot difference list $S_{(j)(j-1)}$
7 comprising a list of one or more data blocks of said j^{th} state snapshot
8 S_j identified as differing from data blocks of said $(j-1)^{\text{th}}$ state
9 snapshot $S_{(j-1)}$;
10 means for copying from said $(j-1)^{\text{th}}$ state snapshot S_{j-1} all the data blocks
11 listed in said $(j-1)^{\text{th}}$ precedent snapshot difference list $S_{(j)(j-1)}$ into a
12 $(j-1)^{\text{th}}$ precedent backup $B_{(j)(j-1)}$; and
13 means for copying said precedent snapshot difference list $S_{(j)(j-1)}$.

1 36. The apparatus of claim 35 further comprising means for storing said $(j-1)^{\text{th}}$
2 precedent backup $B_{(j)(j-1)}$ in an offline memory means.

1 37. The apparatus of claim 36 further comprising:

2 means for retrieving said $(j-1)^{\text{th}}$ precedent backup $B_{(j)(j-1)}$;
 3 means for recovering said $(j-1)^{\text{th}}$ precedent snapshot difference list $S_{(j)(j-1)}$
 4 from said $(j-1)^{\text{th}}$ precedent backup $B_{(j)(j-1)}$; and
 5 means for overwriting at least a portion of said j^{th} state snapshot S_j with at
 6 least a portion of the contents of said $(j-1)^{\text{th}}$ precedent backup
 7 $B_{(j)(j-1)}$.

1 38. The apparatus of claim 37 further comprising:

2 means for generating a $(j-1)^{\text{th}}$ succedent snapshot difference list $S_{(j-2)(j-1)}$
 3 comprising a list of one or more data blocks of said $(j-1)^{\text{th}}$ state
 4 snapshot $S_{(j-1)}$ identified as differing from data blocks of said $(j-2)^{\text{th}}$
 5 state snapshot $S_{(j-2)}$; and
 6 means for copying from said $(j-1)^{\text{th}}$ state snapshot $S_{(j-1)}$ all the data blocks
 7 listed in said $(j-1)^{\text{th}}$ succedent snapshot difference list $S_{(j-2)(j-1)}$ into
 8 said $(j-1)^{\text{th}}$ precedent backup $B_{(j)(j-1)}$; and
 9 means for copying said $(j-1)^{\text{th}}$ succedent snapshot difference list $S_{(j-2)(j-1)}$.

1 39. The apparatus of claim 38 further comprising:

2 means for recovering said $(j-1)^{\text{th}}$ succedent snapshot difference list
 3 $S_{(j-2)(j-1)}$ from said $(j-1)^{\text{th}}$ precedent backup $B_{(j)(j-1)}$; and
 4 means for overwriting at least a portion of said $(j-2)^{\text{th}}$ state snapshot $S_{(j-2)}$
 5 with at least a portion of the contents of said $(j-1)^{\text{th}}$ precedent backup
 6 $B_{(j)(j-1)}$.

1 40. The apparatus of claim 35 further comprising:
2 means for concatenating an $(h-1)^{\text{th}}$ precedent backup $B_{(h)(h-1)}$ with an
3 $(h-2)^{\text{th}}$ precedent backup $B_{(h-1)(h-2)}$ through a g^{th} precedent backup
4 $B_{(g+1)(g)}$, where $g < h$, into a concatenated precedent backup $B_{(h)(g)}$;
5 and
6 means for storing said concatenated precedent backup $B_{(h)(g)}$ in an offline
7 memory means.

1 41. The apparatus of claim 38 further comprising:
2 means for concatenating a $(g+1)^{\text{th}}$ succedent backup $B_{(g)(g+1)}$ with a
3 $(g+2)^{\text{th}}$ succedent backup $B_{(g+1)(g+2)}$ through an h^{th} succedent backup
4 $B_{(h-1)(h)}$, where $g < h$, into a concatenated succedent backup $B_{(g)(h)}$;
5 and
6 means for storing said concatenated succedent backup $B_{(g)(h)}$ in an offline
7 memory means.